

CLAIMS

1. A data-mirroring method comprising:

obtaining control from an I/O process executing on a host computer, the I/O process processing a first I/O request for writing data to a first device;

creating a second I/O request for writing the data to the second device; and

returning control to the I/O process by causing the first I/O request to be made available to a first data storage system managing the first device.
2. The method of claim 1, further comprising determining that the first device is being mirrored by a second device.
3. The method of claim 1, wherein returning control to the I/O process comprises passing control to a DDTSIO module following creation of the second I/O request.
4. The method of claim 1, further comprising

detecting that the first I/O request has been made available to the first data storage system; and,

in response, causing the second I/O request to be provided to a mirror data storage system managing the second device.
5. The method of claim 4, wherein detecting that the first I/O request has been made available to the first data storage system comprises intercepting a response indicating a status of the first I/O request.
6. The method of claim 5, wherein intercepting a response comprises intercepting a condition code indicative of a status of the first I/O request.
7. The method of claim 5, wherein intercepting a response comprises intercepting a return from an IOSVCP module to an IOSVSSCH module.
8. The method of claim 1, wherein obtaining control from an I/O process comprises:

identifying a forward pointer to instructions to be executed by the I/O process in processing the first I/O request; and

causing the forward pointer to point to a front-end detour that includes instructions for creating the second I/O request.

9. The method of claim 1, wherein obtaining control from an I/O process comprises intercepting a call by an IOSVSSCH module to a DDTSIO module.
10. The method of claim 1, wherein obtaining control from an I/O process comprises:
 - identifying a forward pointer that points to instructions for executing a DDTSIO module, and
 - causing the forward pointer to point to instructions for executing a front-end detour, the front-end detour including instructions for creating the second I/O request.
11. The method of claim 1, further comprising:
 - obtaining control from the I/O process after the first I/O request has been made available to a first data storage system managing the first device; and
 - obtaining information indicative of a status of the first I/O request.
12. The method of claim 1, further comprising:
 - identifying a return pointer to a module that is intended to receive information indicative of a status of the first I/O request; and
 - causing the return pointer to point to a back-end detour, the back-end detour including instructions for causing the second I/O request to be provided to a second data storage system managing the second device.
13. The method of claim 12, wherein identifying a return pointer comprises identifying a pointer to an IOSVSSCH module.

14. The method of claim 4, wherein causing the second I/O request to be provided to a second data storage system comprises:

determining whether issuing a request to start an I/O operation is permissible;
and

if starting an I/O operation is not permissible, scheduling the second I/O request
for starting at a later time; and

15. The method of claim 4, wherein causing the second I/O request to be provided to a second data storage system comprises:

determining whether issuing a request to start an I/O operation is permissible;
and

if starting an I/O operation is permissible, starting the second I/O operation.

16. The method of claim 4, further comprising

determining whether both the first I/O request and the second I/O request are
both complete; and

if both the first I/O request and the second I/O request are both complete,
permitting confirmation of completion of the first I/O request; and

if the second I/O request is incomplete, preventing confirmation of completion of
the first I/O request.

17. The method of claim 16, wherein determining whether both the first I/O request
and the second I/O request are complete comprises intercepting information
indicative of status of an I/O request.

18. The method of claim 17, wherein intercepting information indicative of status of
an I/O request comprises:

identifying a status pointer to instructions to be executed by an I/O process in response to the information indicative of status of the I/O request; and

causing the status pointer to point to a post-status detour that includes instructions for

determining whether both the first I/O request and the second I/O request are both complete; and

if both the first I/O request and the second I/O request are both complete, permitting confirmation of completion of the first I/O request; and

if the second I/O request is incomplete, preventing confirmation of completion of the first I/O request.

19. A computer-readable medium having encoded thereon software for executing a data-mirroring computer-readable medium, said software comprising instructions for:

obtaining control from an I/O process executing on a host computer, the I/O process processing a first I/O request for writing data to a first device;

creating a second I/O request for writing the data to the second device; and

returning control to the I/O process by causing the first I/O request to be made available to a first data storage system managing the first device.

20. The computer-readable medium of claim 19, wherein the software further comprises instructions for determining that the first device is being mirrored by a second device; and

21. The computer-readable medium of claim 19, wherein the instructions for returning control to the I/O process comprise instructions for passing control to a DDTSIO module following creation of the second I/O request.

22. The computer-readable medium of claim 19, wherein the software further comprises instructions for:

detecting that the first I/O request has been made available to the first data storage system; and,

in response, causing the second I/O request to be provided to a mirror data storage system managing the second device.
23. The computer-readable medium of claim 22, wherein the instructions for detecting that the first I/O request has been made available to the first data storage system comprise instructions for intercepting a response indicating a status of the first I/O request.
24. The computer-readable medium of claim 23, wherein the instructions for intercepting a response comprise instructions for intercepting a condition code indicative of a status of the first I/O request.
25. The computer-readable medium of claim 23, wherein the instructions for intercepting a response comprise instructions for intercepting a return from an IOSVCP module to an IOSVSSCH module.
26. The computer-readable medium of claim 19, wherein the instructions for obtaining control from an I/O process comprise instructions for:

identifying a forward pointer to instructions to be executed by the I/O process in processing the first I/O request; and

causing the forward pointer to point to a front-end detour that includes instructions for creating the second I/O request.
27. The computer-readable medium of claim 19, wherein the instructions for obtaining control from an I/O process comprise instructions for intercepting a call by an IOSVSSCH module to a DDTSIO module.

28. The computer-readable medium of claim 19, wherein the instructions for obtaining control from an I/O process comprise instructions for:
- identifying a forward pointer that points to instructions for executing a DDTSIO module; and
 - causing the forward pointer to point to instructions for executing a front-end detour, the front-end detour including instructions for creating the second I/O request.
29. The computer-readable medium of claim 19, wherein the software further comprises instructions for instructions for:
- obtaining control from the I/O process after the first I/O request has been made available to a first data storage system managing the first device; and
 - obtaining information indicative of a status of the first I/O request.
30. The computer-readable medium of claim 19, wherein the software further comprises instructions for:
- identifying a return pointer to a module that is intended to receive information indicative of a status of the first I/O request; and
 - causing the return pointer to point to a back-end detour, the back-end detour including instructions for causing the second I/O request to be provided to a second data storage system managing the second device.
31. The computer-readable medium of claim 30, wherein the instructions for identifying a return pointer comprise instructions for identifying a pointer to an IOSVSSCH module.
32. The computer-readable medium of claim 22, wherein the instructions for causing the second I/O request to be provided to a second data storage system comprise instructions for:

determining whether issuing a request to start an I/O operation is permissible;
and

if starting an I/O operation is not permissible, scheduling the second I/O request
for starting at a later time; and

33. The computer-readable medium of claim 22, wherein the instructions for causing the second I/O request to be provided to a second data storage system comprise instructions for:

determining whether issuing a request to start an I/O operation is permissible;
and

if starting an I/O operation is permissible, starting the second I/O operation.

34. The computer-readable medium of claim 22, wherein the software further comprises instructions for instructions for:

determining whether both the first I/O request and the second I/O request are
both complete; and

if both the first I/O request and the second I/O request are both complete,
permitting confirmation of completion of the first I/O request; and

if the second I/O request is incomplete, preventing confirmation of completion of
the first I/O request.

35. The computer-readable medium of claim 34, wherein the instructions for determining whether both the first I/O request and the second I/O request are complete comprise instructions for intercepting information indicative of status of an I/O request.

36. The computer-readable medium of claim 35, wherein the instructions for intercepting information indicative of status of an I/O request comprise instructions for:

identifying a status pointer to instructions to be executed by an I/O process in response to the information indicative of status of the I/O request; and

causing the status pointer to point to a post-status detour that includes instructions for

determining whether both the first I/O request and the second I/O request are both complete; and

if both the first I/O request and the second I/O request are both complete, permitting confirmation of completion of the first I/O request; and

if the second I/O request is incomplete, preventing confirmation of completion of the first I/O request.